AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (currently amended) A method of managing a data buffer ~~(120)~~ comprising a queue of consecutive segments of data packets in a base station system (100) of a mobile communications system ~~(1)~~, comprising the steps of:

- said base station system ~~(100)~~ comparing a size ~~(S(k))~~ of a data packet segment ~~(P(k))~~ with a size ~~(S(k+1))~~ of a next consecutive data packet segment ~~(P(k+1))~~ in said buffer ~~(120)~~;
- said base station system ~~(100)~~ identifying said complete data packet based on said comparison; and
- said base station system ~~(100)~~ discarding said identified complete data packet from said buffer ~~(120)~~.

2. (currently amended) The method according to claim 1, wherein said identifying step comprises the steps of:

- identifying said next data packet segment ~~(P(k+1))~~ as a first data packet segment ~~(P(FIRST))~~ of said complete data, packet in said buffer ~~(120)~~ if said size ~~(S(k))~~ of said data packet segment ~~(P(k))~~ is smaller than said size ~~(S(k+1))~~ of said next data packet segment ~~(S(k+1))~~; and
- associating said identified first data packet segment ~~(P(FIRST))~~ with a first segment identifier ~~(FIRST)~~.

3. (currently amended) The method according to claim 1, wherein said identifying step comprises the steps of:

- identifying said next data packet segment ~~(P(k+1))~~ as a last data packet segment ~~(P(LAST))~~ of said complete data packet in said buffer ~~(120)~~ if said size ~~(S(k))~~ of said data packet segment ~~(P(k))~~ differs from said size ~~(S(k+1))~~ of said next data packet segment ~~(P(k+1))~~; and
- associating said identified last data packet segment ~~(P(LAST))~~ with a last segment identifier ~~(LAST)~~.

4. (currently amended) The method according to claim 2 ~~and 3~~, wherein said discarding step comprises the step of discarding said data, packet segment ~~(P(FIRST))~~ associated with said first segment identifier ~~(FIRST)~~, said data packet segment ~~(P (LAST))~~ associated with said last segment identifier ~~(LAST)~~ and any intermediate data packet segments between said data packet segment ~~(P(FIRST))~~ associated with said first segment identifier ~~(FIRST)~~ and said data packet segment ~~(P(LAST))~~ associated with said last segment identifier ~~(LAST)~~ in said buffer ~~(120)~~.

5. (currently amended) A system ~~(130)~~ for managing a data buffer ~~(120) comprising~~ including a queue of consecutive segments of data packets in a base station system ~~(100)~~ of a mobile communications system ~~(1), comprising~~ electronic circuitry configured to:

- ~~means (146) for comparing~~ compare a size ~~(S(k))~~ of a data packet segment ~~(P(k))~~ with a size ~~(S(k+1))~~ of a next consecutive data packet segment ~~(P(k+1))~~ in said buffer ~~(120)~~;
- ~~means (140) for identifying~~ identify said complete data packet based on said comparison; and
- ~~means (136) for discarding~~ discard said identified complete data packet from said

buffer (120).

6. (currently amended) The system according to claim 5, wherein said ~~identifying means (140) is adapted for identifying~~ electronic circuitry is configured to identify said next data packet segment ~~(P(k+ 1))~~ as a first data packet ~~(P(FIRST))~~ of said complete data packet in said buffer ~~(120)~~ if said size ~~(S(k))~~ of said data packet segment ~~(P(k))~~ is smaller than said size ~~(S(k+ 1))~~ of said next data packet segment ~~(P(k+ 1))~~, said system ~~(130)~~ further comprises means ~~(142)~~ for associating said identified first data packet segment ~~(P(FIRST))~~ with a first segment identifier ~~(FIRST)~~.

7. (currently amended) The system according to claim 5, wherein said ~~identifying means (140) is adapted for identifying~~ electronic circuitry is configured to identify said next data packet segment (P(k+ 1)) as a last data packet segment (P(LAST)) of said complete data packet in said buffer (120) if said size (S(k)) of said data packet segment (P(k)) differs from said size (S(k+ 1)) of said next data packet segment (P(k+ 1)), said system (130) further comprises means (142) for associating said identified last data packet segment (P(LAST)) with a last segment identifier (LAST).

8. (currently amended) The system according to claim 6, wherein said ~~discarding means (136) is adapted for discarding~~ electronic circuitry is configured to discard said data packet segment ~~(P(FIRST))~~ associated with said first segment identifier ~~(FIRST)~~, said data packet segment ~~(P(LAST))~~ associated with said last segment identifier ~~(LAST)~~ and any intermediate data packet segments between said data packet segment ~~(P(FIRST))~~ associated with said first

segment identifier ~~(FIRST)~~ and said data packet segment ~~(P(LAST))~~ associated with said last segment identifier ~~(LAST)~~ in said buffer ~~(120)~~.

9. (currently amended) A base station network node of a base station system (100) in a mobile communications system ~~(1)~~ comprising:

- a data buffer ~~(120)~~ comprising a queue of consecutive segments of data packets; and
- a system ~~(130)~~ for managing said data buffer ~~(120)~~ according to -claim 5.

10. (currently amended) A method of enabling identification of a complete data packet in a data buffer ~~(120)~~ comprising a queue of consecutive data packet segments, comprising the steps of:

- comparing a size ~~(S(k))~~ of a data packet segment ~~(P(k))~~ with a size ~~(S(k+1))~~ of a next consecutive data packet segment ~~(P(k+1))~~ in said buffer ~~(120)~~; and
- identifying said complete data packet based on said comparison.

11. (currently amended) The method according to claim 10, further comprising the step of providing a segment counter (k) associated with a data packet segment ~~(P(k))~~ in said buffer ~~(120)~~.

12. (currently amended) The method according to claim 11, further comprising the steps of:

- comparing a size ~~(S(k))~~ of said data packet segment ~~(P(k))~~ associated with said counter ~~(k)~~ with a size ~~(S(k+1))~~ of a next consecutive data packet segment ~~(P(k+1))~~ in said buffer ~~(120)~~;

- identifying said next data packet segment ~~(P(k+1))~~ as a first data packet segment ~~(P(FIRST))~~ of said complete data packet in said buffer ~~(120)~~ if said size ~~(S(k))~~ of said data packet segment ~~(P(k))~~ associated with said counter ~~(k)~~ is smaller than said size ~~(S(k+1))~~ of said next data packet segment ~~(P(k+1))~~.

13. (currently amended) The method according to claim 11, further comprising the steps of:

(a) comparing a size ~~(S(k))~~ of the data packet segment ~~(P(k))~~ currently associated with said counter ~~(k)~~ with a size ~~(S(k+1))~~ of a next consecutive data packet segment ~~(P(k+1))~~ in said buffer ~~(120)~~; and

(b) associating said counter ~~(k)~~ with said next data packet segment ~~(P(k+1))~~ if said size ~~(S(k))~~ of the data packet segment ~~(P(k))~~ currently associated with said counter ~~(k)~~ is equal to or larger than said size ~~(S(k+1))~~ of said next data packet segment ~~(P(k+1))~~; and

- repeating both said comparison step (a) and said associating step (b) until said size ~~(S(k))~~ of the data packet ~~(P(k))~~ currently associated with said counter (k) is smaller than said size ~~(S(k+1))~~ of said next data packet segment ~~(P(k+1))~~, whereby said next data packet segment ~~(P(k+1))~~ is identified as a first data packet segment ~~(P(FIRST))~~ of said complete data packet in said buffer ~~(120)~~.

14. (currently amended) The method according to claim 12, further comprising the step of associating said segment counter ~~(k)~~ with said first data packet segment ~~(P(FIRST))~~ of said complete data packet.

15. (currently amended) The method according to claim 14, further comprising the steps of:

- comparing a size ~~(S(k))~~ of said data packet segment ~~(P(k))~~ associated with said counter ~~(k)~~ with a size ~~(S(k+1))~~ of a next consecutive data packet segment ~~(P(k+1))~~ in said buffer ~~(120)~~; and
- identifying said next data packet segment ~~(P(k+1))~~ as a last data packet segment ~~(P(LAST))~~ of said complete data packet in said buffer ~~(120)~~ if said size ~~(S(k))~~ of said data packet segment ~~(P(k))~~ associated with said counter ~~(k)~~ differs from said size ~~(S(k+1))~~ of said next data packet segment ~~(P(k+1))~~.

16. (currently amended) The method according to claim 15, wherein said complete data packet is identified as comprising said first data packet segment ~~(P(FIRST))~~ of said complete data packet, said last data packet segment ~~(P(LAST))~~ of said complete data packet and any intermediate data packet segments between said first ~~(P(FIRST))~~ and last ~~(P(LAST))~~ data packet segment of said complete data packet in said buffer ~~(120)~~.

17. (currently amended) The method according to claim 15, further comprising the steps of:

- determining a total size of said first data packet segment ~~(P(FIRST))~~ of said complete data packet, said last data packet segment ~~(P(LAST))~~ of said complete data packet and any intermediate data packet segments between said first ~~(P(FIRST))~~ and last ~~(P(LAST))~~ data packet segment of said complete data packet in said buffer ~~(120)~~;
- comparing said total size with a minimum size threshold; and
- identifying said complete data packet as comprising said first data packet segment ~~(P(FIRST))~~ of said complete data packet, said last data packet segment ~~(P(LAST))~~ of said

complete data packet and any intermediate data packet segments between said first ~~(P(FIRST))~~ and last ~~(P(LAST))~~ data packet segment of said complete data packet in said buffer ~~(120)~~ if said total size is larger than said minimum size threshold.

18. (currently amended) The method according to claim 11, further comprising the steps of:

- comparing a size ~~(S(k))~~ of said data packet segment ~~(P(k))~~ associated with said counter ~~(k)~~ with a size ~~(S(k+1))~~ of a next consecutive data packet segment ~~(P(k+1))~~ in said buffer ~~(120)~~; and

- identifying said next data packet segment ~~(P(k+1))~~ as a last data packet segment ~~(P(LAST))~~ of said complete data packet in said buffer ~~(120)~~ if said size ~~(S(k))~~ of said data packet segment ~~(P(k))~~ associated with said counter ~~(k)~~ differs from said size ~~(S(k+1))~~ of said next data packet segment ~~(P(k+1))~~.

19. (currently amended) The method according to claim 11, further comprising the steps of:

(c) comparing a size ~~(S(k))~~ of the data packet segment ~~(P(k))~~ currently associated with said counter ~~(k)~~ with a size ~~(S(k+1))~~ of a next consecutive data packet segment ~~(P(k+1))~~ in said buffer ~~(120)~~;

(d) associating said counter ~~(k)~~ with said next data packet segment ~~(P(k+1))~~ if said size ~~(S(k))~~ of the data packet segment ~~(P(k))~~ currently associated with said counter ~~(k)~~ is equal to said size ~~(S(k))~~ of said next data packet segment ~~(P(k+1))~~; and

- repeating both said comparison step (c) and said associating step (d) until said size ~~(S(k))~~ of the data packet segment ~~(P(k))~~ currently associated with said counter ~~(k)~~ differs from said size ~~(S(k+1))~~ of said next data packet segment ~~(P(k+1))~~, whereby said next data packet

segment ~~(P(k+1))~~ is identified as a last data packet segment ~~(P(LAST))~~ of said complete data packet in said buffer ~~(120)~~.

20. (currently amended) A system ~~(140)~~ for enabling identification of a complete data packet in a data buffer ~~(120)~~ comprising a queue of consecutive data packet segments, comprising:

- means ~~(146)~~ for comparing a size ~~(S(k))~~ of a data packet segment ~~(P(k))~~ with a size ~~(S(k+1))~~ of a next consecutive data packet segment ~~(P(k+1))~~ in said buffer ~~(120)~~; and
- means ~~(140)~~ for identifying said complete data packet based on said comparison.

21. (currently amended) The system according to claim 20, comprising means ~~(142)~~ for associating a segment counter ~~(k)~~ with a data packet segment ~~(P(k))~~ in said buffer ~~(120)~~.

22. (currently amended) The system according to claim 21, wherein said comparison means ~~(146)~~ is adapted for comparing a size ~~(S(k))~~ of said data packet segment ~~(P(k))~~ associated with said counter ~~(k)~~ with a size ~~(S(k+1))~~ of a next consecutive data packet segment ~~(P(k+1))~~ in said buffer ~~(120)~~, wherein said identifying means ~~(140)~~ is adapted for identifying said next data packet segment ~~(P(k+1))~~ as a first data packet segment ~~(P(FIRST))~~ of said complete data packet in said buffer ~~(120)~~ if said size ~~(S(k))~~ of said data packet segment ~~(P(k))~~ associated with said counter ~~(k)~~ is smaller than said size ~~(S(k+1))~~ of said next data packet segment ~~(P(k+1))~~.

23. (currently amended) The system according to claim 21, wherein said comparison means ~~(146)~~ is adapted for comparing a size ~~(S(k))~~ of the data packet segment ~~(P(k))~~ currently

associated with said counter ~~(k)~~ with a size ~~(S(k+1))~~ of a next consecutive data packet segment ~~(P(k+1))~~ in said buffer ~~(120)~~, wherein said associating means ~~(142)~~ is adapted for associating said counter ~~(k)~~ with said next data packet segment ~~(P(k+1))~~ if said size ~~(S(k))~~ of the data packet segment ~~(P(k))~~ currently associated with said counter ~~(k)~~ is equal to or larger than said size ~~(S(k+1))~~ of said next data packet segment ~~(P(k+1))~~, said comparison means ~~(146)~~ is adapted for repeating said size comparison and said associating means ~~(142)~~ is adapted for repeating said counter association until said size ~~(S(k))~~ of the data packet segment ~~(P(k))~~ currently associated with said counter ~~(k)~~ is smaller than said size ~~(S(k+1))~~ of said next data packet segment ~~(P(k+1))~~, whereby said identifying means ~~(140)~~ is adapted for identifying said next data packet segment ~~(P(k+1))~~ as a first data packet segment ~~(P(FIRST))~~ of said complete data packet in said buffer ~~(120)~~.

24. (currently amended) The system according to claim 22, wherein said associating means ~~(142)~~ is adapted for associating said segment counter ~~(k)~~ with said first data packet segment ~~(P(FIRST))~~ of said complete data packet.

25. (currently amended) The system according to claim 24, wherein said comparison means ~~(146)~~ is adapted for comparing a size ~~(S(k))~~ of said data packet segment ~~(P(k))~~ associated with said counter ~~(k)~~ with a size ~~(S(k+1))~~ of a next consecutive data packet segment ~~(P(k+1))~~ in said buffer ~~(120)~~, wherein said identifying means ~~(140)~~ is adapted for identifying said next data packet segment ~~(P(k+1))~~ as a last data packet segment ~~(P(LAST))~~ of said complete data packet in said buffer ~~(120)~~ if said size ~~(S(k))~~ of said data packet segment ~~(P(k))~~ associated with said counter ~~(k)~~ differs from said size ~~(S(k+1))~~ of said next data packet segment (P(k+ 1)).

26. (currently amended) The system according to claim 25, wherein said identifying means ~~(140)~~ is adapted for identifying said complete data packet as comprising said first data packet segment ~~(P(FIRST))~~ of said complete data packet, said last data packet segment ~~(P(LAST))~~ of said complete data packet and any intermediate data packet segments between said first ~~(P(FIRST))~~ and last ~~(P(LAST))~~ data packet segment of said complete data packet in said buffer ~~(120)~~.

27. (currently amended) The system according to claim 25, further comprising means ~~(142)~~ for determining a total size of said first data packet segment ~~(P(FIRST))~~ of said complete data packet, said last data packet segment ~~(P(LAST))~~ of said complete data packet and any intermediate data packet segments between said first ~~(P(FIRST))~~ and last ~~(P(LAST))~~ data packet segment of said complete data packet in said buffer ~~(120)~~, said comparison means ~~(146)~~ is adapted for comparing said total size with a minimum size threshold, and said identifying means ~~(140)~~ is adapted for identifying said complete data packet as comprising said first data packet segment ~~(P(FIRST))~~ of said complete data packet, said last data packet segment ~~(P(LAST))~~ of said complete data packet and any intermediate data packet segments between said first ~~(P(FIRST))~~ and last ~~(P(LAST))~~ data packet segment of said complete data packet in said buffer ~~(120)~~ if said total size is larger than said minimum size threshold.

28. (currently amended) The system according to claim 21, wherein said comparison means ~~(146)~~ is adapted for comparing a size ~~(S(k))~~ of said data packet segment ~~(P(k))~~ associated with said counter ~~(k)~~ with a size ~~(S(k+1))~~ of a next consecutive data packet segment ~~(P(k+1))~~ in said

buffer ~~(120)~~, wherein said identifying means ~~(140)~~ is adapted for identifying said next data packet segment ~~(P(k+1))~~ as a last data packet segment ~~(P(LAST))~~ of said complete data packet in said buffer ~~(120)~~ if said size ~~(S(k))~~ of said data packet segment ~~(P(k))~~ associated with said counter ~~(k)~~ differs from said size ~~(S(k+1))~~ of said next data packet segment ~~(P(k+1))~~.

29. (currently amended) The system according to claim 21, wherein said comparison means ~~(146)~~ is adapted for comparing a size ~~(S(k))~~ of the data packet segment ~~(P(k))~~ currently associated with said counter ~~(k)~~ with a size ~~(S(k+1))~~ of a next consecutive data packet segment ~~(P(k+1))~~ in said buffer ~~(120)~~, wherein said associating means ~~(142)~~ is adapted for associating said counter ~~(k)~~ with said next data packet segment ~~(P(k+1))~~ if said size ~~(S(k))~~ of the data packet segment ~~(P(k))~~ currently associated with said counter ~~(k)~~ is equal to said size ~~(S(k+1))~~ of said next data packet segment ~~(P(k+1))~~, said comparison means ~~(146)~~ is adapted for repeating said size comparison and said associating means ~~(142)~~ is adapted for repeating said counter associating unxxxxtil said size ~~(S(k))~~ of the data packet segment ~~(P(k))~~ currently associated with said counter ~~(k)~~ differs from said size ~~(S(k+1))~~ of said next data packet segment ~~(P(k+1))~~, whereby said identifying means ~~(140)~~ is adapted for identifying said next data packet segment ~~(P(k+1))~~ as a last data packet segment ~~(P(LAST))~~ of said complete data packet in said buffer ~~(120)~~.